Computer Architecture

SJTU — 2022 Spring

519021911248 *Zhuohao Li* *Instructor: Prof. Yanyan Shen*

Homework 3

§ Memory Hierarchy §

Problem 1: Principle of Locality

You are trying to appreciate how important the principle of locality is in justifying the use of a cache memory, so you experiment with a computer having an L1 data cache and a main memory (you exclusively focus on data accesses). The latencies (in CPU cycles) of the different kinds of accesses are as follows: cache hit, 1 cycle; cache miss, 105 cycles; main memory access with cache disabled, 100 cycles.

(1) When you run a program with an overall miss rate of 5%, what will the average memory access time (in CPU cycles)be? .

Average memory access time=(1-miss rate)× hit time + miss rate × miss time
$= 0.95 \times 1 + 0.05 \times 105 = 6.2$ cycles.

(2) Next, you run a program specifically designed to produce completely random data addresses with no locality. Toward that end, you use an array of size 256 MB (all of it fits in the main memory). Accesses to random elements of this array are continuously made(using a uniform random number generator to generate the elements indices). If your data cache size is 64 KB, what will the average memory access time be? .

Accesses are produced randomly with no locality. So the cache rate is:

$$r = \frac{64KB}{256MB} = 0.00025$$

Average memory access time= $0.00025 \times 1 + (1 - 0.00025) \times 105 = 104.974$ cycles

(3) If you compare the result obtained in part (b) with the main memory access time when the cache is disabled, what can you conclude about the role of the principle of locality in justifying the use of cache memory? .

The access time when the cache is disabled is 100 cycles, which is less than the average access time(104.974) when the cache is enabled and almost all the accesses are random and missed. If there is no locality at all in the data stream then the cache memory will not only be useless but also it will be a liability.

(4) You observed that a cache hit produces a gain of 99 cycles(1 cycle vs.100), but it produces a loss of 5 cycles in the case of a miss(105 cycles vs.100). In the general case, we can express these two quantities as G(gain) and L(loss). Using these two quantities (G and L), identify the highest miss rate after which the cache use would be disadvantageous. .

Assuming the memory access time with no cache is T_{off}, with cache is T_{on}. Miss rate is r. Then the average access time is:

$$T_{on} = (1 - r)(T_{off} - G) + r(T_{off} + L)$$

To make it efficient and profitable, we should satisfied:

$$T_{off} \geq (1 - r)(T_{off} - G) + r(T_{off} + L)$$

We have $r \leq (\frac{G}{G + L}) = 99/104$

Problem 2: Average Memory Access Time

You are building a system around a processor with in-order execution that runs at 1.1 GHz and has a CPI of 0.7 excluding memory accesses. The only instructions that read or write data from memory are loads(20%of all instructions) and stores(5%of all instructions).
The memory system for this computer is composed of a split L1 cache that imposes no penalty on hits. Both the l-cache and D-cache are direct mapped and hold 32 KB each. The l-cache has a 2% miss rate and 32-byte blocks, and the D-cache is write through with a 5%miss rate and 16- byte blocks. There is a write buffer on the D-cache that eliminates stalls for 95%of all writes.
The 512 KB write-back, unified L2 cache has 64-byte blocks and an access time of 15 ns. It is connected to the L1 cache by a 128-bit data bus that runs at 266 MHz and can transfer one 128-bit word per bus cycle. Of all memory references sent to the L2 cache in this system, 80% are satisfied without going to main memory. Also, 50%of all blocks replaced are dirty.
The 128-bit-wide main memory has an access latency of 60 ns, after which any number of bus words may be transferred at the rate of one per cycle on the 128-bit-wide 133 MHz main memory bus.

First some notes:

Symbol	Meaning	Value
CR	Clock Rate	$1.1GHz$, cycle time $1/(1.1\times10^9)=0.9ns$
$CPIEM$	CPI excluding memory accesses	$0.7cycles$
L	Loads	20%
S	Store	5%
L1$		
$T_{HIT,I\$}$	Time penalty for hit in I$, L1	0.
$T_{HIT,D\$}$	Time penalty for hit in D$, L1	0
$\%miss_{I\$}$	miss rate for I$	0.02
$\%miss_{D\$}$	miss rate for D$	0.05
$B_{L1:I\$}$	L1 I$ Block	$32B/block$
$B_{L1:D\$}$	L1 D$ Block	$16B/block$
C_{L1}	L1 Capacity	$32KB$
L2 $		
C_{L2}	L2 Capacity	$512KB$
B_{L2}	L2 Block	$64B/block$
$t_{HIT,L2}$	Accesstime in L2	$15ns$
BUS_{L2}	L2 Data bus	$128bit$, $\frac{64\times8}{128}=4$times transformation for one block
T_{L2}	L2 128bit transfer cycle	$266MHz$
t_{L2}	L2 unit transfer time	$\frac{1}{266\times10^6}=3.76ns$
$\%miss_{L2}$	miss rate in L2	20%
DR_{L2}	dirty rate in L2	50%
Main Memory		
w	width	$128bits$
$t_{AL,MM}$	Access latency for main memory	$60ns$
t_{MM}	Main memory transfer time	$\frac{1}{133\times10^6}=7.52ns$
$t_{LW,MM}$	load/write block latency	$60ns+4\times7.52=90ns$

(1) What is the average memory access time for instruction accesses?

$$\begin{aligned}AWAT &= T_{HIT,I\$} + \%miss, I\$ \times (T_{HIT,L2} + Fetch\_Miss\_Block\_from\_L2 + Fetch\_Miss\_Block\_from\_MM \\ &\quad + Write\_Back\_the\_Dirty\_Block\_When\_Miss) \\ &= 0 + 0.02 \times (15 + \frac{32 \times 8}{128} \times 3.76 + 0.2 \times (60 + \frac{64 \times 8}{128} \times 7.52) + 0.2 \times 0.5 \times (60 + \frac{64 \times 8}{128} \times 7.52)) \\ &= 0.99ns\end{aligned}$$

(2) What is the average memory access time for data reads?

$$\begin{aligned}AWAT &= T_{HIT,D\$} + \%miss, D\$ \times (T_{HIT,L2} + Fetch\_Miss\_Block\_from\_L2 + Fetch\_Miss\_Block\_from\_MM \\ &\quad + Write\_Back\_the\_Dirty\_Block\_When\_Miss) \\ &= 0 + 0.05 \times (15 + \frac{16 \times 8}{128} \times 3.76 + 0.2 \times (60 + \frac{64 \times 8}{128} \times 7.52) + 0.2 \times 0.5 \times (60 + \frac{64 \times 8}{128} \times 7.52)) \\ &= 2.29ns\end{aligned}$$

(3) What is the average memory access time for data writes? .

Every write will go to store in write buffer. Each time for one word.

$$\begin{aligned}AWAT &= T_{HIT,L1,buffer} + Stall\_rate \times (t_{HIT,L2} + \%miss_{L2} \times Miss\_penalty\_of\_L2) \\ &= 0 + 0.05 \times (15 + \times 3.76 + 0.2 \times (60 + 7.52) \\ &= 1.61ns\end{aligned}$$

If considering cache replacement due to write allocation

$$\begin{aligned}AWAT &= 0 + 0.05 \times (15 + \times 3.76 + 0.2 \times (90 + 50\% \times 90) \\ &= 2.29ns\end{aligned}$$

(4) What is the overall CPl,including memory accesses?

$$\begin{aligned}CPI &= origin + stalls\_for\_instruction + stalls\_for\_read + stalls\_for\_write \\ &= 0.7 + 0.99/0.0 + 20\% \times 2.29/0.9 + 5\% \times 2.29/0.9 \\ &= 2.44\end{aligned}$$

Problem 3: Process of Accessing

A program is running on a computer with a four-entry fully associative (micro) translation lookaside buffer (TLB):

VP#	PP#	Entry valid
5	30	1
7	1	0
10	10	1
15	25	1

Virtual page index	Physical page#	Present
0	0	Y
1	7	N
2	6	N
3	5	Y
4	14	Y
5	30	Y
6	26	Y
7	11	Y
8	13	N
9	18	N
10	10	Y
11	56	Y
12	110	Y
13	33	Y
14	12	N
15	25	Y

The following is a trace of virtual page numbers accessed by a program. For each access indicate whether it produces a TLB hit/miss and, if it accesses the page table, whether it produces a page hit or fault. Put an X under the page table column if it is not accessed.

The result table is:

Virtual page index	Physical page#	Present
1	miss	fault
5	hit	/
9	miss	fault
14	miss	fault
10	hit	/
6	miss	hit
15	hit	/
12	miss	hit
7	miss (7# was replaced by 6#)	hit
2	miss	fault